LISTING OF AND AMENDMENTS TO CLAIMS:

1. (currently amended) In a computing system that includes dynamic compilation capability, a method for executing a computer program including controlling the execution of an instruction of the computer program, comprising the steps of:

translating an instruction from a first representation to a translated representation, and setting a tag associated with the instruction in the first representation; and

prior to execution of a given instruction in the first representation, examining the tag associated with the given instruction, and if such associated tag has been set, branching to the translated version of the given instruction, for further execution of the program; and if said tag has not been set, interpreting and compiling the given instruction from said first representation, for further execution of the program;

wherein said examining of said tag is effected without first attempting ~~performing~~ a cache fetch, so that the time required to attempt a cache fetch is saved if said tag has not been set.

2. (original) The method of claim 1, comprising the further step of looking up the address of the translated version of the instruction.

3. (original) The method of claim 2, further comprising executing the translated version of the given instruction, and, upon reaching the end of such execution, determining

whether a subsequent instruction is to be executed, and if so, determining whether such subsequent instruction exists in a translated version by examining a tag associated with a first representation of the subsequent instruction.

4. (original) The method of claim 3, wherein the translated version is stored in cache memory.

5. (original) The method of claim 4, wherein the translated instruction is an optimized version of the first instruction, but in the same instruction set as the first instruction.

6. (original) The method of claim 4, wherein the translated instruction is represented in a different instruction set than the first instruction.

7. (previously presented) The method of claim 1, wherein the tag is a single bit.

8. (original) The method of claim 1, wherein the tag is represented by a field of multiple bits, said field also indicating information about the instruction in the first representation selected from the group consisting of: profile information and exception information.

9. (original) The method of claim 1, wherein a single tag corresponds to a plurality of instructions in the first representation.

10. (canceled)

11. (currently amended) In a computing system that executes a computer program, and includes dynamic compilation capability of the computer program, the improvement comprising:

means for translating an instruction from a first representation of the program to a translated representation, and for setting a tag associated with the instruction in the first representation;

means for, prior to execution of a given instruction in the first representation, examining, without first attempting ~~performing~~ a cache fetch, so that the time required to attempt a cache fetch is saved, the tag associated with the given instruction, and if such associated tag has been set, branching to the translated version of the given instruction, for further execution of the program; and

means for interpreting and compiling the given instruction from said first representation, for further execution of the program, if said tag has not been set.

12. (original) The system of claim 11, further comprising means for looking up the address of the translated version of the instruction.

13. (original) The system of claim 12, further comprising means for executing the translated version of the given instruction, and, upon reaching the end of such execution, determining whether a subsequent instruction is to be executed, and if so, determining whether such subsequent instruction exists in a translated version by examining a

tag associated with a first representation of the subsequent instruction.

14. (original) The system of claim 13, wherein the translated version is stored in cache memory.

15. (original) The system of claim 14, wherein the translated instruction is an optimized version of the first instruction, but in the same instruction set as the first instruction.

16. (original) The method of claim 14, wherein the translated instruction is represented in a different instruction set than the first instruction.

17. (previously presented) The system of claim 11, wherein the tag is a single bit.

18. (original) The system of claim 11, wherein the tag is represented by a field of multiple bits, said field also indicating information about the instruction in the first representation selected from the group consisting of: profile information and exception information.

19. (original) The system of claim 11, wherein a single tag corresponds to a plurality of instructions in the first representation.

20. (canceled).

21. (previously presented) The system of claim 11, further comprising an exception handler for execution of the program when an exception in said translated representation occurs.